

**In the Claims:**

1. (Original) A method for multicasting data comprising:
  - adding the data to a queue;
  - placing a reference to the data and to a multicast tree into a to-do list;
  - for each of the leaves of the multicast tree, obtaining a data element pointer and a leaf to be processed pointer from the to-do list; and
  - scheduling the data to be output to each of the leaves of the multicast tree based on the data element pointer and the leaf to be processed pointer.
2. (Original) The method of claim 1 further comprising:
  - updating the leaf to be processed pointer in the to-do list for each instance of scheduling the data to be output to each of the leaves.
3. (Original) The method of claim 2 further comprising:
  - setting a copy count to an initial value;
  - adjusting the copy count based on a number of actual copies made.
4. (Original) The method of claim 3 wherein the step of setting the copy count to the initial value further comprises:
  - setting the copy count to a maximum value initially, and
  - wherein the step of adjusting the copy count further comprises:
    - decrementing the copy count for each of the actual copies made.
5. (Original) The method of claim 3 wherein the step of adjusting the copy count is performed in response to the step of scheduling the data to be output to each of the leaves of the multicast tree.
6. (Original) The method of claim 4 wherein the step of scheduling the data to be output to each of the leaves of the multicast tree further comprises:
  - counting every one of the leaves of the multicast tree.

7. (Original) The method of claim 4 wherein the step of adjusting the copy count is performed based on reaching a leaf to be processed pointer that points to itself.

8. (Original) A method for multicasting data comprising:
- adding the data to a queue as enqueued data;
  - transmitting the enqueued data to each of a plurality of multicast destinations;
  - determining that all of the enqueued data for a particular destination of the multicast destinations have been transmitted.

9. (Original) A method for multicasting data comprising:
  - storing queue context information and data context information pertaining to the data;
  - determining which of the data to transmit as output data and a destination to which it shall be transmitted based on the queue context information and the data context information;
  - transmitting the output data to the destination;
  - determining a number of destinations to which the output data has been transmitted; and
  - determining a number of destinations intended to receive the output data.
10. (Original) The method of claim 9 further comprising:
  - when the number of destinations to which the output data has been transmitted indicates that the output data has been transmitted to the number of destinations intended to receive the output data, concluding the queuing of the output data.
11. (Original) The method of claim 10 further comprising:
  - updating the queue context information for a root connection.
12. (Original) The method of claims 11 further comprising:
  - obtaining a data element pointer and a leaf to be processed pointer from a to-do list, wherein the step of determining which of the data to transmit as the output data is based on the data element pointer and the leaf to be processed pointer.
13. (Original) The method of claim 12 further comprising:
  - updating a number of leaves counter.
14. (Original) The method of claim 12 further comprising:
  - adjusting a copy count, wherein the copy count is used to determine the number of destinations to which the output data has been transmitted.
15. (Original) The method of claim 14 wherein the step of adjusting the copy count further comprises:
  - decrementing the copy count for each of the number of destinations to which the output data has been transmitted, wherein the copy count had been initialized to a maximum value.

16. (Original) Apparatus for multicasting data comprising:
- a queuing element for receiving the data and for transmitting the data to a plurality of multicast destinations via a plurality of destination queues;
  - a memory subsystem for storing the data, queue context information pertaining to a root connection, data context information pertaining to the data, a data element pointer, and a leaf to be processed pointer; and
  - a scheduler for scheduling a dequeuing event for the data.
17. (Original) The apparatus of claim 16 wherein the memory subsystem comprises:
- a data memory for storing the data;
  - a queue context block for storing queue context information for each of the destination queues;
  - a data context block for storing data context information for each of the data elements of the data; and
  - a to-do list for storing the data element pointer and the leaf to be processed pointer.
18. (Original) The apparatus of claim 17 wherein the data context information comprises:
- a copy count for counting events of transmitting the data; and
  - a next data element pointer for indicating a next data element of the data to be processed.
19. (Original) The apparatus of claim 18 wherein the queue context information comprises:
- a head pointer;
  - a tail pointer; and
  - a root pointer.
20. (Original) The apparatus of claim 19 wherein the to-do list further comprises:
- a leaf count for counting destination queues to which the data shall be transmitted.